Journal of Dynamics, Monitoring and Diagnostics, 2025, 4, 170-182
https://doi.org/10.37965/jdmd.2025.855

ISTP
RESEARCH ARTICLE

Robust Anomaly Detection of Rotating Machinery with Contaminated Data

Jingcheng Wen,[1,2] Jiaxin Ren,[1,2] Zhibin Zhao,[1,2] and Xuefeng Chen[1,2]
[1]National Key Lab of Aerospace Power System and Plasma Technology, Xi'an Jiaotong University, Xi'an, P.R. China
[2]School of Mechanical Engineering, Xi'an Jiaotong University, Xi'an, P.R. China

(Received 30 June 2025; Revised 15 July 2025; Accepted 15 August 2025; Published online 18 August 2025)

Abstract: Rotating machinery is critical to industrial systems, necessitating robust anomaly detection (AD) to ensure operational safety and prevent failures. However, in real-world scenarios, monitoring data is typically unlabeled and often consists of normal samples contaminated with a small proportion of unknown anomalies. To address this, this paper proposes a diffusion-based AD method, Anomaly Detection Denoising Diffusion Probabilistic Model (AD-DDPM) for robust AD. The method employs a U-attention-net to capture local and global features and introduces a filtered contrastive mechanism to mitigate the impact of contaminated training data. By leveraging the probabilistic nature of diffusion models, AD-DDPM effectively models normal data distributions, achieving superior AD even with polluted samples. Experimental validation on fault simulation datasets demonstrates the method's exceptional performance, outperforming traditional machine learning and deep learning baselines. The proposed approach offers a promising solution for reliable health monitoring in industrial settings.

Keywords: anomaly detection; contaminated data; diffusion model; rotating machinery

I. INTRODUCTION

Rotating machinery is a cornerstone of modern industry, integral to complex systems such as aero-engines, wind turbines, tracked vehicles, and nuclear pumps. Ensuring the operational safety of these systems and preventing catastrophic failures require real-time monitoring of the machinery's health status [1]. Anomaly detection (AD), as the initial step in such monitoring systems, identifies deviations from normal operational behavior and alerts users to potential issues [2]. Therefore, developing accurate and robust AD methods is essential for reliable equipment performance and timely maintenance.

A straightforward approach to AD involves setting a threshold for a health index and triggering an alarm when this threshold is exceeded [3]. However, extracting a precise health index and determining an appropriate threshold pose significant challenges. To address these limitations, machine learning algorithms like k-means clustering [4] and one-class support vector machines (OCSVMs) [5] have been proposed for AD. Nevertheless, these methods often rely on distance-based similarity measures, which become less effective in high-dimensional spaces due to the curse of dimensionality. Moreover, they typically lack the ability to capture temporal dependencies and dynamic patterns inherent in time-series data. Deep learning has recently gained prominence as it extracts complex features layer by layer through neural networks, achieving superior performance.

Deep AD methods falls into three broad classes: supervised, semi-supervised, and unsupervised. Supervised AD treats AD as a classification task, relying on labeled data to learn the distinction between normal and anomalous instances. Semi-supervised AD combines limited labeled anomalies with abundant unlabeled samples to train models, while unsupervised AD relies solely on normal data for training. In industrial settings, where monitoring data is often unlabeled, unsupervised AD is particularly suitable. Among unsupervised AD methods, reconstruction-based approaches using autoencoders are particularly popular [6]. Autoencoders are trained on unlabeled data, employing an encoder–decoder architecture to reconstruct input signals. During testing, a signal is fed into the model, and a large reconstruction error indicates a potential anomaly.

Despite the advancements in deep learning-based AD, particularly with reconstruction-based methods like autoencoders, several challenges persist in industrial applications such as polluted data and noise attack [7]. Autoencoders assume that normal data can be accurately reconstructed, while anomalies yield high reconstruction errors. However, in real-world industrial environments, unlabeled anomalies frequently exist within the training data. It degrades the model performance as autoencoders tend to learn detailed pattern from point-to-point reconstructions and overfit the anomalies.

To model the monitoring long-term signals, generative modeling approaches such as diffusion models may be a promising method for unsupervised AD. Diffusion models operate by iteratively adding noise in a forward process and recover the data distribution in a reverse phase. The unique mechanism allows diffusion models to learn robust representations of normal data, even in the presence of contaminated samples, by modeling the data distribution in a probabilistic manner.

In particular, the signals from rotating machinery often contain complex temporal dependencies, periodic patterns, and subtle deviations. These characteristics pose significant challenges for conventional autoencoders, which typically perform pointwise reconstruction and are sensitive to contamination in the training data. Diffusion models, by

Corresponding author: Zhibin Zhao (e-mail: zhaozhibin@xjtu.edu.cn).

contrast, leverage a probabilistic generative framework capable of capturing long-term structure and denoising capabilities, making them more suitable to detect weak anomalies in such industrial scenarios.

In this paper, a diffusion-based method for robust AD in the presence of contaminated samples is proposed. It contains two parts. First, a Denoising Diffusion Probabilistic Model (DDPM) is trained to learn a robust representation of signals. Second, the signals are corrupted by iterative noise addition over fixed steps, followed by a denoising process that reconstructs an approximation of the healthy signal. Furthermore, to address the challenge of contaminated samples, we designed a filtered contrastive mechanism (FCM) for robust AD. To summarize, the key contributions are as follows:

1. A diffusion-based AD method for signals is proposed, incorporating a U-attention-net to jointly capture local patterns and global dependencies.
2. A FCM for AD is presented to enhance robustness against the contaminated samples. Pseudo-label filtering is followed by a contrastive penalty that pulls together similar features while pushing apart anomalies.
3. The method is validated on two parts-level and components-level fault simulation datasets and demonstrates its superiority.

The article is organized into five sections: a review of existing literature in Section II, the details of the proposed methodology in Section III, experiments and result analysis in Section IV, and concluding remarks in Section V.

II. RELATED WORKS

A. ANOMALY DETECTION FOR MACHINERY

Methods for machine AD include statistical, machine learning, and deep learning approaches. Statistical ones extract health indicator from time-domain feature [8], frequency domain feature [9], or entropy-based feature [10] and apply rules such as 3-sigma threshold to identify anomalies. Machine learning methods can be categorized into distance- and density-based methods. The former, such as K-nearest neighbors (KNN) [11], OCSVM [12], and SVDD [13] calculate the distances between data points, assuming the anomalies are relatively far from normal points. The latter, including Local Outlier Factor (LOF) [14] and Isolation Forest (IF) [15], assume that the normal data clusters densely while the anomalies occupy low-density regions.

Deep AD methods in machinery are categorized into reconstruction-based and adversarial-based ones. Reconstruction-based methods primarily rely on autoencoders, which consist of an encoder–decoder architecture. Li *et al.* [16] constructed a convolutional autoencoder incorporating dilated casual convolution and skip connection for gearbox AD. Yang *et al.* [17] introduced a behavior- and condition-aware variational autoencoder framework, with the reconstruction errors used to identify gearbox failure. Adversarial-based methods employ a generator to produce normal data and a discriminator to distinguish whether abnormal samples are consistent with the generated distribution [18].

Diffusion-based AD is prevalent in fields like medical imaging [19] and video analysis [20]. Diffusion-based models have also been researched for time series. Recent literature such as D^3R [21], ImDiffusion [22], and DDMT [23] has successfully adapted diffusion-based approaches for multivariate time-series AD, with demonstrated advantages in robustness to drift, reconstruction fidelity, and anomaly scoring accuracy. However, in the field of PHM, diffusion-based models are mostly employed to generate new samples in fault diagnosis [24], with limited application to AD. The proposed method implicitly models long-term time series from a diffusion perspective, enabling more accurate and robust AD.

B. ANOMALY DETECTION ON CONTAMINATED DATA

Handling contaminated training data is a critical challenge in AD, and a common strategy involves refining the training dataset. Yoon *et al.* [25] removed the anomalies in the polluted dataset by an ensemble of one-class classifiers. The samples predicted as normal by all classifiers are retained in the refined data. Ulmer *et al.* [26] divided data into overlapping subsets to train an ensemble of models, assigning refinement scores based on the contribution of samples. The method was validated on AD of machine audio recordings and aeroengine sensor data. Du *et al.* [27] filtered potential anomalies in time-series data by comparing generated signals and original signals, enabling the discriminator to focus on normal patterns. Shang *et al.* [28] considered the essential self-clean characteristic of autoencoders and designed a weighted gradient updating strategy to prioritize core samples for AD in acoustic signals in machines and pressure signals in gear pumps.

Some methods enhance performance by extracting information from both normal and anomalous data. For example, Latent Outlier Exposure (LOE) proposed by Qiu *et al.* [29] infers pseudo-labels of samples and jointly optimizes normal and anomalous data via two loss functions with shared model parameters. Mou *et al.* [30] integrated contrastive learning with one-class classification, treating original and reconstructed time series as positive pairs while introducing an outlier exposure term that pushes anomalous samples away by reversing the objective for abnormal data. Su *et al.* [31] proposed CIBiGAN, which distinguishes between normal samples, generated samples, and anomalies by leveraging contaminated data to better characterize the normal data distribution. In addition, other approaches have been explored, such as combining robust principal component analysis (RPCA) with autoencoders [32] or employing self-supervised frameworks to identify outliers without requiring clean labels [33].

The proposed method integrates data refinement and the joint utilization of both normal and abnormal information within a contrastive learning framework, aiming to achieve more robust and effective AD.

III. PROPOSED METHOD

To realize robust AD in the polluted training data, the method consists of two parts: the AD DDPM (AD-DDPM) and the FCM as shown in Fig. 1. AD-DDPM provides the steps for AD in signals. The FCM processes polluted input data for accurate prediction.

DDPM [34] is a classical diffusion model for generating samples. It includes two stages: forward process and reverse process.

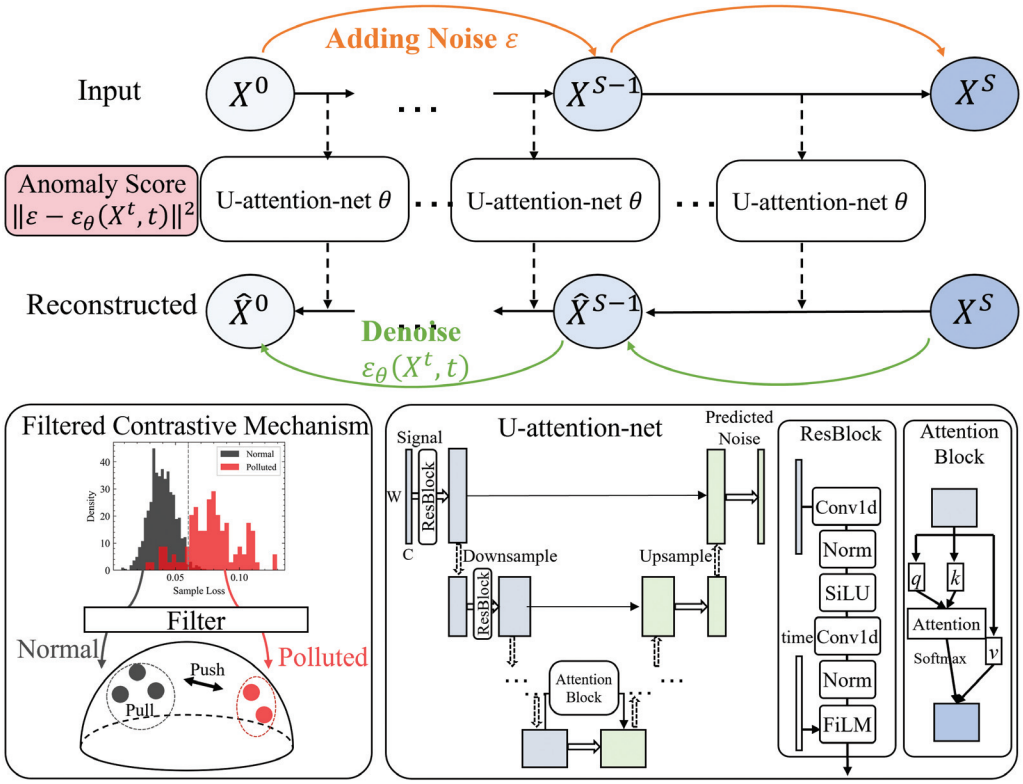

Fig. 1. Proposed AD-DDPM method for robust anomaly detection for rotating machines.

A. FORWARD PROCESS

During the forward process, Gaussian noise is incrementally added to the input signal across several steps, as illustrated in Fig. 2. The noise strength at each step is determined by a predefined scheduler. The forward step t produces X^t by perturbing the previous signal X^{t-1} with Gaussian noise, thereby establishing a Markov chain where X^t depends solely on X^{t-1} and is not influenced by the steps before. Given the Gaussian noise $\varepsilon \sim \mathcal{N}(0,\mathbf{I})$ with variance controlled by β_t, the recursive version of the forward process is as follows:

$$X^t = \sqrt{1-\beta_t}X^{t-1} + \sqrt{\beta_t}\varepsilon \tag{1}$$

where β_t follows a schedule and ε is drawn from a standard Gaussian distribution. Defining $\alpha_t = 1 - \beta_t$, the forward process can be rewritten as:

$$X^t = \sqrt{\alpha_t}X^{t-1} + \sqrt{1-\alpha_t}\varepsilon \tag{2}$$

X^t is a combination of X^{t-1} and added noise ε. By iterating the process, the signal at step t can be expressed relative to the original data x^0:

$$X^t = \sqrt{\bar{\alpha}_t}X^0 + \sqrt{1-\bar{\alpha}_t}\varepsilon \tag{3}$$

where $\bar{\alpha}_t = \prod_{i=1}^{t} \alpha_i$. At the final step T, the signal X^T is distributed according to a standard Gaussian $\mathcal{N}(0,\mathbf{I})$.

B. REVERSE PROCESS

The reverse process, depicted in Fig. 2, seeks to denoise the signal and recover the original data distribution from the noised output of the forward process. In line with the Markov chain property, X^{t-1} is predicted conditionally on X^t. The conditional distribution is given by:

$$P(X^{t-1}|X^t,X^0) = \frac{P(X^t|X^{t-1},X^0)P(X^{t-1}|X^0)}{P(X^t|X^0)} \tag{4}$$

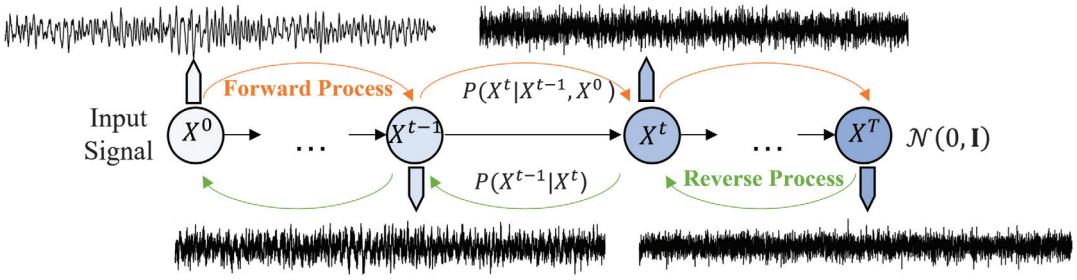

Fig. 2. Forward noise addition and reverse denoising process in DDPM.

From the forward process, the distributions are as follows:

$$\begin{aligned} P(X^t|X^{t-1},X^0) &\sim \mathcal{N}(\sqrt{\alpha_t}X^{t-1},\sqrt{1-\alpha_t}\mathbf{I}) \\ P(X^t|X^0) &\sim \mathcal{N}(\sqrt{\overline{\alpha_t}}X^0,\sqrt{1-\overline{\alpha_t}}\mathbf{I}) \\ P(X^{t-1}|X^0) &\sim \mathcal{N}(\sqrt{\overline{\alpha_{t-1}}}X^0,\sqrt{1-\overline{\alpha_{t-1}}}\mathbf{I}) \end{aligned} \tag{5}$$

Thus, the distribution $P(x^{t-1}|x^t,x^0)$ can be derived from equations (4) and (5):

$$\begin{aligned} P(X^{t-1}|X^t,X^0) &\sim \mathcal{N}(\mu^{t-1},\sigma^{t-12}\mathbf{I}) \\ \mu^{t-1} &= \frac{\sqrt{\alpha_t}(1-\bar{\alpha}_{t-1})X^t + \sqrt{\bar{\alpha}_{t-1}}(1-\alpha_t)X^0}{1-\bar{\alpha}_t} \\ \sigma^{t-12} &= \frac{(1-\alpha_t)(1-\bar{\alpha}_{t-1})}{1-\bar{\alpha}_t} \end{aligned} \tag{6}$$

Using equation (3), $X^0 = \frac{1}{\sqrt{\bar{\alpha}_t}}(X^t - \sqrt{1-\bar{\alpha}_t}\varepsilon)$, the mean can be reformulated as:

$$\mu^{t-1} = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(X^t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon\right) \tag{7}$$

Thus, the reverse distribution becomes

$$P(X^{t-1}|X^t) \sim \mathcal{N}\left(\frac{1}{\sqrt{\bar{\alpha}_t}}\left(X^t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon\right), \frac{(1-\alpha_t)(1-\overline{\alpha}_{t-1})}{1-\overline{\alpha}_t}\right) \tag{8}$$

The noise term $\varepsilon \sim \varepsilon_\theta(x_t,t)$ is approximated by a neural network θ. During the training stage, Gaussian noise is added at step t, with the loss computed as the mean square error between the true noise $\varepsilon \sim \mathcal{N}(0,\mathbf{I})$ and the predicted noise $\varepsilon_\theta(X^t,t)$:

$$L(\theta) = \mathbb{E}_{t,X^0,\varepsilon}\left[\|\varepsilon - \varepsilon_\theta(\sqrt{\bar{\alpha}_t}X^0 + \sqrt{1-\bar{\alpha}_t}\varepsilon,t)\|^2\right] \tag{9}$$

After training, the backward sampling process can be achieved using the well-trained network. The recursive version of the reverse process is

$$X^{t-1} = \frac{1}{\sqrt{\bar{\alpha}_t}}\left(X^t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\varepsilon_\theta(X^t,t)\right) + \sigma^t z \tag{10}$$

where $z \sim \mathcal{N}(0,\mathbf{I})$ is random Gaussian noise.

C. AD-DDPM

A dataset of time series with N samples $D = \{X_1,X_2,\ldots,X_N\}$ is given, where each sample $X_i = \{x_1,x_2,\ldots x_W\}$ is a signal with length W. $x_w \in \mathbb{R}^M$ represents a timestamp at time w with channels M. Each sample X_i is associated with an unknown label $y_i \in \{0,1\}$, where 0 represents normal and 1 denotes anomalous. The training dataset is assumed to be contaminated, with a contamination ratio $\sigma = N_2/(N_1+N_2)$, where N_1 and N_2 represent the counts of normal and anomalous samples, respectively.

During training, input samples are processed by adding Gaussian noise as part of the forward process of a DDPM. By minimizing the loss function defined in equation (9), the network learns to predict noise and approximate the distribution of normal samples, enabling robust representation of normal signals.

In the testing phase, an input sample is corrupted through the forward process over S fixed steps, yielding a noised signal X_i^S. The anomaly score is defined as the difference between the predicted noise $\varepsilon_\theta(X^S,S)$ at step S and the actual noise ε:

$$Score = |\varepsilon_\theta(X^S,S) - \varepsilon| \tag{11}$$

Samples with an anomaly score exceeding a predefined threshold δ are classified as anomalous.

In this paper, a new model structure for predicting noise called U-attention-net is proposed to enhance the denoising capabilities of AD-DDPM. As depicted in Fig. 1, the network architecture follows a U-shaped encoder–decoder structure [35] that utilizes dilated convolutional neural network (CNN) and self-attention to capture local details and global dependencies from input time-series signals, enabling robust representation learning for normal samples.

Time Step Conditioning: The forward step is embedded by a sinusoidal positional encoding scheme of dimension 128, followed by a two-layer multilayer perceptron (MLP) with hidden size 128 and SiLU activation. The embedding vector modulates the features through Feature-wise Linear Modulation (FiLM) layers [36], which generate per-channel scaling and bias terms applied after group normalization in encoder.

Encoder: It is composed of four downsampling stages. Each stage contains a residual block with two one-dimensional convolutional layers of kernel size 3, followed by group normalization with eight groups and SiLU activation. Dilation factors increase exponentially across the stages, taking values of 1, 2, 4, and 8, which allows the receptive field to expand. After each residual block, the temporal dimension is reduced by a factor of 2 through average pooling. The number of feature channels doubles from stage to stage, progressing from 64 in the first block to 128, 256, 512, and finally 1024 channels at the bottleneck input.

Attention Bottleneck: It is inspired by the attention mechanism in transformers [37]. A 1D convolutional layer projects the feature map into query Q, key K, and value V, each of dimension $C = 1024$. The attention score is the scaled dot-product $\text{Attention}(Q,K,V) = \text{softmax}(QK^T/\sqrt{C})V$. The result is projected back via another 1×1 convolution before being added to the block input through a residual connection, enabling the model to capture long-range dependencies and augment the local patterns learned by convolutions.

Decoder: The decoder mirrors the encoder in reverse. Each stage begins with nearest-neighbor interpolation that doubles the temporal dimension, after which the upsampled feature map is combined with the corresponding skip connection from the encoder through element-wise addition. A residual block then processes the merged features, reducing the channel dimension from 1024 down to 64. The dilation factors in the decoder decrease in reverse order, moving from 8 in the first stage down to 1 in the final stage. A final one-dimensional convolution with kernel size 3 maps the output of the last decoder block to the channel of input signals.

D. FILTERED CONTRASTIVE MECHANISM

To deal with the challenge of contaminated training data, a FCM is proposed, comprising the part of filtering and a contrastive learning penalty.

In real-world industrial settings, training datasets often contain a mixture of normal and anomalous samples, with the latter typically being less prevalent. During preliminary

training, the model tends to fit the normal samples more, prioritizing inliers' loss function [28,33]. Consequently, the anomaly score for normal samples is generally lower than for anomalies. Leveraging this property, we set a threshold based on the $(1-\sigma)$ quantile of the loss distribution. Samples with losses exceeding this threshold, corresponding to the σ-percentage of the dataset, are assigned pseudo-labels as anomalous, while the remainder is labeled as normal. After each training epoch, the losses from the noise estimation network at a fixed diffusion step are collected to update these pseudo-labels:

$$L_{con} = -\log\left(\frac{\sum_{i,j\in\{D_n\}}\exp\left(\frac{sim(z_i,z_j)}{\tau}\right)}{\sum_{i,j\in\{D_n\}}\exp\left(\frac{sim(z_i,z_j)}{\tau}\right)+\sum_{i\in\{D_n\},k\in\{D_a\}}\exp\left(\frac{sim(z_i,z_k)}{\tau}\right)}\right) \tag{12}$$

Following the filtering process, a contrastive learning penalty is introduced to further improve the model's discrimination between normal and abnormal ones. The contrastive mechanism operates on the feature representations extracted after the U-Net bottleneck, denoted as z_i. The goal is to pull the feature representations of pseudo-normal samples D_n closer together while pushing them away from those pseudo-anomalous samples D_a. The contrastive loss is defined in equation (12) where $sim(z_i,z_j) = \frac{z_i * z_j}{\|z_i\|\|z_j\|}$ represents the cosine similarity between feature vectors, and τ is a temperature hyperparameter that controls the softness of the similarity distribution. This loss encourages the model to learn feature representations to cluster normal samples tightly in the latent space, while pushing anomalies apart, enhancing the model's discriminative power. By integrating this contrastive penalty with the DDPM loss, the model achieves greater robustness against contaminated data, as it learns to focus on the core patterns of normal samples while marginalizing the influence of anomalies.

In summary, Algorithm 1 illustrates the training and inference processes.

IV. EXPERIMENTS

This section presents experimental validation of the proposed method through simulations at both the part level and the component level. One is a SQI planetary gearbox transmission test rig, referred to as SQI, which focuses on localized faults, and the other is a helicopter main gearbox test platform, referred to as MGB, which reflects system-level faults. The experiments utilized an NVIDIA RTX 3090 GPU, running Python 3.9 and PyTorch 2.7.

Algorithm 1. AD-DDPM

Train phase

Given dataset $D = \{X_1, X_2, \ldots, X_N\}$ with pollution σ. Initialize DDPM noise scheduler β, U-attention-net with weights w, sample mask m (all ones), trade-off parameter λ

Repeat

Split data $D_n = m \odot D$ $D_a = (1-m) \odot D$

Sample noise $\varepsilon \sim \mathcal{N}(0,1)$

Forward $X^t = \sqrt{\bar{\alpha}_t} X^0 + \sqrt{1-\bar{\alpha}_t}\varepsilon$

Loss $L(\theta) = \|\varepsilon - \varepsilon_\theta(X^t, t)\|^2 + \lambda \cdot L_{con}$

Optimize parameters w

Update mask m with percentage $1-\sigma$

Until convergence

Inference phase

Given a sample X

Sample noise $\varepsilon \sim \mathcal{N}(0,1)$

Forward at S step $X^S = \sqrt{\bar{\alpha}_S} X^0 + \sqrt{1-\bar{\alpha}_S}\varepsilon$

Compute Score = $|\varepsilon_\theta(X^S, S) - \varepsilon|$

Label=abnormal if Score > threshold else normal

A. DATASET DESCRIPTION

SQI test rig comprises a motor, a controller, a brake, a two-stage planetary gearbox, and a two-stage fixed-shaft gearbox, as illustrated in Fig. 3(a). Vibration and acoustic signals were captured using an acceleration sensor and a microphone, respectively. To simulate anomalous conditions, a sun gear tooth break fault was introduced by replacing the component in the test rig. The normal and tooth break sun gears are shown in Fig. 3(b) and (c). Both vibration and acoustic signals under normal and tooth break conditions were collected at 51,200 Hz. The rotating speed was set to 3000 rpm and the load brake current was set to 0.4A.

MGB test rig, as shown in Fig. 4(a), is composed of drive motor, load motor, lubrication and cooling system, and main gearbox. Figure 4(b) illustrates that the main gearbox consists of a bevel gear stage, a spur gear stage, and a planetary gear stage. A gear ring crack in the planetary gear stage, as shown in Fig. 4(c), is introduced as the simulated fault for anomaly analysis. Both vibration and acoustic signals are collected with sampling rate 51,200Hz.

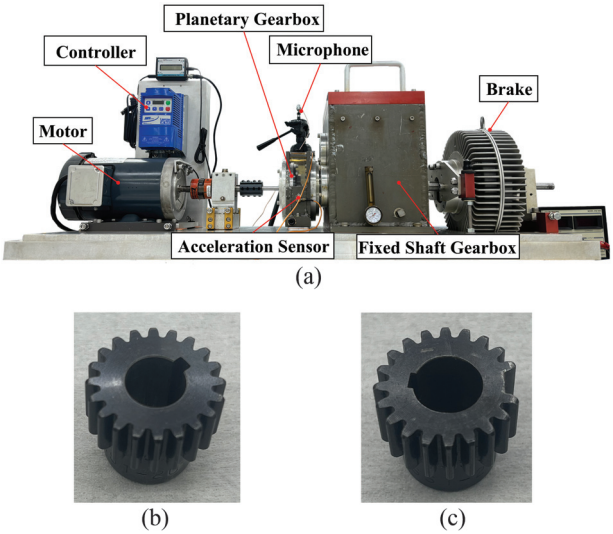

Fig. 3. Experimental platform and the sun gear components. (a) SQI planetary gearbox transmission test rig. (b) Normal sun gear. (c) Tooth break sun gear.

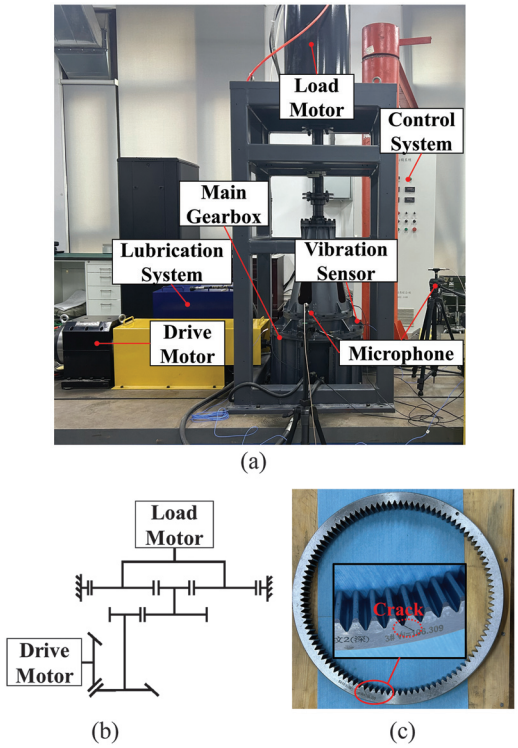

Fig. 4. Experimental platform and the gear ring tooth crack fault component. (a) MGB helicopter main gearbox test platform. (b) Transmission chain of the main gearbox. (c) Tooth crack fault of the gear ring.

The operating condition is set to an input speed of 3000 rpm with no additional load.

B. EXPERIMENTAL SETUP

The Adam optimizer was employed for 200 training epochs with a learning rate of 0.0001. The samples are extracted using a sliding window of size 5120 and step 2560. The vibration and acoustic signals are integrated at the data level. They are stacked along the channel dimension to form a two-channel input matrix of shape 5120×2, which preserves the raw temporal correspondence between the two modalities. To simulate data contamination, the ratio of fault samples to normal samples in the training and validation datasets is denoted as σ. Sliding windows are first applied to generate all samples. For normal conditions, we divide the data in a non-overlapping manner according to the time order, with the first 80% used for training and the remaining 20% for testing. For fault conditions, a number of fault samples equal to $\sigma/(1–\sigma)$ times the number of normal training samples are added to the training set to achieve a contamination ratio of σ. The remaining fault samples are used for testing. The test set is sampled to maintain a 1:1 ratio between normal and fault samples. Specifically, in the SQI dataset, the training set contains 927 normal samples along with a corresponding number of contaminated fault samples determined by σ. The test set includes 232 normal and 232 fault samples. In the MGB dataset, the training set consists of 479 normal samples and corresponding contaminated fault samples. The test set includes 120 normal samples and 120 fault samples.

The data is normalized with the mean $\mu(x_{:,m})$ and standard deviation $\sigma(x_{:,m})$ along each channel m:

$$x'_{w,m} = \frac{x_{w,m} - \mu(x_{:,m})}{\sigma(x_{:,m})} \tag{13}$$

The noise scheduler used in the DDPM follows a linear schedule with 500 steps, while the step S in the testing process is 100. The temperature τ for contrastive penalty is 0.5 and the trade-off coefficient is 0.1. To alleviate the influence of randomness, each model is trained five times with different seeds. The reported results include both the mean and standard deviation.

C. EVALUATION METRICS

The effectiveness of the proposed method is assessed using several metrics: area under curve (AUC), accuracy (Acc), F1 score (F1), true positive rate (TPR), and false alarm rate (FAR). AUC, the area under the ROC curve, provides a threshold-independent measure of the model's discriminative capability. The other metrics depend on a specific threshold. In this experiment, the threshold was determined by evaluating anomaly scores for all samples and selecting the one that maximizes the F1 score. The metrics are defined as follows:

$$\text{Acc} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FP} + \text{TN} + \text{FN}} \tag{14}$$

$$\text{F1} = 2 \cdot \frac{\frac{\text{TP}}{\text{TP}+\text{FP}} \cdot \frac{\text{TP}}{\text{TP}+\text{FN}}}{\frac{\text{TP}}{\text{TP}+\text{FP}} + \frac{\text{TP}}{\text{TP}+\text{FN}}} \tag{15}$$

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{16}$$

$$\text{FAR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \tag{17}$$

Higher values indicate better performance for AUC, Acc, F1, and TPR, while lower values are better for FAR.

D. PERFORMANCE COMPARISON

The proposed approach is benchmarked against machine learning methods (OCSVM [38], IF [39], LOF [40]) and deep learning methods, autoencoder (AE), and variational autoencoder (VAE) [41]. Three autoencoder architectures are implemented, namely CNN-AE [28], GRU-AE [42], and LSTM-AE [27]. Experiments are implemented under three contamination levels ($\sigma = 0$, 0.1, 0.2) to assess the robustness of each method to polluted training data.

The evaluation metrics of SQI dataset are summarized in Table I, with the best performance for each metric highlighted in bold. The proposed AD-DDPM model consistently outperforms all baseline methods across all metrics and noise levels, demonstrating its robustness and efficacy in AD. The machine learning methods exhibit poor performance for AD with low accuracy and high FAR. These methods struggle to deal with high-dimensional data and fail to capture complex temporal dependencies. The reconstruction-based autoencoders, regardless of their architecture or whether they use a variational framework (VAE), outperform most traditional machine learning approaches. However, their performance remains suboptimal compared to that of AD-DDPM. Specifically, compared with the next

Table I. Performance comparison on the SQI dataset: the proposed method versus machine learning and deep learning approaches across various evaluation metrics under different contamination ratios

	AUC(↑)	Acc(↑)	F1(↑)	TPR(↑)	FAR(↓)
			$\sigma = 0$		
OCSVM	0.65954 ± 0.00000	61.422% ± 0.000%	70.724% ± 0.000%	92.241% ± 0.000%	69.397% ± 0.000%
IF	0.57351 ± 0.01409	57.457% ± 0.995%	69.334% ± 0.325%	95.259% ± 1.329%	80.345% ± 3.147%
LOF	0.86151 ± 0.00000	77.155% ± 0.000%	80.300% ± 0.000%	91.810% ± 0.000%	37.500% ± 0.000%
AE-CNN	0.91675 ± 0.01679	85.560% ± 1.648%	85.493% ± 1.689%	85.560% ± 1.648%	20.259% ± 4.385%
AE-GRU	0.90033 ± 0.00696	82.155% ± 1.155%	82.092% ± 1.182%	82.155% ± 1.155%	22.155% ± 4.747%
AE-LSTM	0.88796 ± 0.00657	81.422% ± 0.992%	81.185% ± 0.915%	81.422% ± 0.992%	29.655% ± 1.681%
VAE	0.94571 ± 0.01011	87.414% ± 1.814%	87.407% ± 1.808%	87.414% ± 1.814%	13.965% ± 2.012%
AD-DDPM	**1.00000 ± 0.00000**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**0.000% ± 0.000%**
			$\sigma = 0.1$		
OCSVM	0.69345 ± 0.00000	60.345% ± 0.000%	70.813% ± 0.000%	95.259% ± 0.000%	74.569% ± 0.000%
IF	0.61736 ± 0.00000	57.112% ± 0.000%	69.725% ± 0.000%	97.845% ± 0.000%	83.621% ± 0.000%
LOF	0.88563 ± 0.00000	79.741% ± 0.000%	81.729% ± 0.000%	89.224% ± 0.000%	29.741% ± 0.000%
AE-CNN	0.89532 ± 0.01380	83.491% ± 0.883%	83.451% ± 0.879%	83.491% ± 0.883%	21.293% ± 1.242%
AE-GRU	0.88266 ± 0.01169	81.509% ± 0.894%	81.492% ± 0.881%	81.509% ± 0.894%	20.690% ± 2.301%
AE-LSTM	0.85903 ± 0.00675	79.655% ± 0.657%	79.524% ± 0.673%	79.655% ± 0.657%	28.276% ± 1.542%
VAE	0.92049 ± 0.01468	84.871% ± 1.658%	84.838% ± 1.670%	84.871% ± 1.658%	12.328% ± 3.263%
AD-DDPM	**0.99958 ± 0.00028**	**99.353% ± 0.305%**	**99.353% ± 0.305%**	**99.353% ± 0.305%**	**0.603% ± 0.385%**
			$\sigma = 0.2$		
OCSVM	0.68271 ± 0.00000	62.500% ± 0.000%	71.215% ± 0.000%	91.810% ± 0.000%	66.810% ± 0.000%
IF	0.62238 ± 0.00000	59.267% ± 0.000%	69.968% ± 0.000%	93.966% ± 0.000%	75.431% ± 0.000%
LOF	0.88548 ± 0.00000	79.957% ± 0.000%	81.890% ± 0.000%	89.224% ± 0.000%	29.310% ± 0.000%
AE-CNN	0.88163 ± 0.01531	81.078% ± 1.164%	81.047% ± 1.186%	81.078% ± 1.164%	20.086% ± 4.328%
AE-GRU	0.85713 ± 0.01250	78.879% ± 0.927%	78.767% ± 0.930%	78.879% ± 0.927%	27.069% ± 4.890%
AE-LSTM	0.82682 ± 0.00678	76.897% ± 1.060%	76.730% ± 1.089%	76.897% ± 1.060%	31.465% ± 1.928%
VAE	0.89889 ± 0.01179	83.017% ± 1.315%	82.992% ± 1.315%	83.017% ± 1.315%	13.448% ± 2.076%
AD-DDPM	**0.99911 ± 0.00052**	**98.879% ± 0.515%**	**98.879% ± 0.515%**	**98.879% ± 0.515%**	**1.379% ± 1.029%**

best model, VAE, our proposed AD-DDPM gets better accuracy in percentages of 14.4%, 17.1%, and 19.1% for pollution ratios 0, 0.1, and 0.2. As noise levels increase from 0 to 0.2, AD-DDPM maintains high performance across all metrics, with only marginal degradation. F or instance, AUC drops slightly from 1.00000 to 0.99911, reflecting robust generalization even under polluted training conditions. In contrast, baseline models exhibit more pronounced performance declines. AE's accuracy drops from 85.560% to 81.078% and VAE's AUC decreases from 0.94571 to 0.89889.

Table II presents the performance comparison on the MGB dataset under varying levels of label noise. The proposed AD-DDPM model consistently achieves top scores across all evaluation metrics and pollution ratios, demonstrating strong resilience to noisy supervision. Even as the contamination ratio increases from 0 to 0.2, the degradation in performance remains marginal, with AUC only slightly reduced from 1.00000 to 0.98507 and FAR increasing modestly from 0.000% to 4.167%. This stability confirms the model's ability to generalize well under challenging conditions. In contrast, traditional machine learning methods such as OCSVM and IF completely fail across all settings, yielding near-zero AUC, indicating their inability to handle complex, high-dimensional time-series data. Although LOF initially performs well when the dataset is clean, its performance collapses rapidly as noise increases, with AUC dropping to 0.33056 and then to 0.31090 under 0.1 and 0.2 noise ratios, respectively. AE models offer moderate improvements over classical baselines but are hindered by high FARs and unstable detection performance, particularly under higher noise levels. For instance, AE-CNN's FAR increases to over 50% under a 0.2 pollution ratio. VAE achieves competitive results when the training data is clean, but its robustness is limited. As noise is introduced, its performance deteriorates noticeably, with AUC falling to 0.93335 and FAR rising to 12.667% at 0.2 pollution. Compared to VAE, AD-DDPM consistently outperforms across all metrics and remains significantly more robust to data contamination.

The superior performance of AD-DDPM across all metrics and noise levels highlights its potential for AD with imperfect data. It achieves near-perfect detection rates while maintaining extremely low false positives, making it suitable for industrial monitoring, where both high sensitivity and precision are critical.

E. ABLATION STUDY

To assess the contribution of the U-attention-net for extracting comprehensive information from high-dimension data, we conduct an ablation study by removing its key

Table II. Performance comparison on the MGB dataset: the proposed method versus machine learning and deep learning approaches across various evaluation metrics under different contamination ratios

	AUC(↑)	Acc(↑)	F1(↑)	TPR(↑)	FAR(↓)
			$\sigma = 0$		
OCSVM	0.00000 ± 0.00000	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
IF	0.00021 ± 0.00019	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
LOF	0.99701 ± 0.00000	97.500% ± 0.000%	97.942% ± 0.000%	98.333% ± 0.000%	3.333% ± 0.000%
AE-CNN	0.70325 ± 0.07937	69.083% ± 6.630%	67.974% ± 6.889%	69.083% ± 6.630%	48.333% ± 8.720%
AE-GRU	0.72229 ± 0.03517	68.500% ± 3.418%	68.328% ± 3.367%	68.500% ± 3.418%	38.333% ± 3.281%
AE-LSTM	0.73799 ± 0.00952	70.167% ± 1.369%	69.975% ± 1.201%	70.167% ± 1.369%	36.500% ± 4.346%
VAE	0.99990 ± 0.00014	99.750% ± 0.373%	99.750% ± 0.373%	99.750% ± 0.373%	0.333% ± 0.456%
AD-DDPM	**1.00000 ± 0.00000**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**0.000% ± 0.000%**
			$\sigma = 0.1$		
OCSVM	0.00000 ± 0.00000	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
IF	0.00028 ± 0.00000	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
LOF	0.33056 ± 0.00000	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
AE-CNN	0.62788 ± 0.02537	62.333% ± 1.807%	62.135% ± 1.610%	62.333% ± 1.807%	40.333% ± 7.327%
AE-GRU	0.69852 ± 0.03532	66.833% ± 2.926%	66.754% ± 2.982%	66.833% ± 2.926%	36.500% ± 6.021%
AE-LSTM	0.70761 ± 0.00416	68.417% ± 0.801%	68.164% ± 0.708%	68.417% ± 0.801%	40.333% ± 1.395%
VAE	0.98954 ± 0.00781	96.417% ± 1.807%	96.413% ± 1.811%	96.417% ± 1.807%	3.667% ± 2.923%
AD-DDPM	**0.99999 ± 0.00003**	**99.917% ± 0.186%**	**99.917% ± 0.186%**	**99.917% ± 0.186%**	**0.167% ± 0.373%**
			$\sigma = 0.2$		
OCSVM	0.00000 ± 0.00000	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
IF	0.00056 ± 0.00000	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
LOF	0.31090 ± 0.00000	49.583% ± 0.000%	66.667% ± 0.000%	99.167% ± 0.000%	100.000% ± 0.000%
AE-CNN	0.58150 ± 0.03364	59.583% ± 2.483%	59.080% ± 2.492%	59.583% ± 2.483%	51.500% ± 2.312%
AE-GRU	0.65182 ± 0.03101	63.250% ± 2.251%	62.999% ± 2.440%	63.250% ± 2.251%	37.667% ± 8.066%
AE-LSTM	0.65983 ± 0.00429	64.583% ± 1.179%	64.444% ± 1.177%	64.583% ± 1.179%	40.667% ± 3.604%
VAE	0.93335 ± 0.04380	88.667% ± 5.356%	88.654% ± 5.367%	88.667% ± 5.356%	12.667% ± 6.439%
AD-DDPM	**0.98507 ± 0.00920**	**95.500% ± 1.873%**	**95.499% ± 1.874%**	**95.500% ± 1.873%**	**4.167% ± 2.041%**

Table III. Evaluation metrics of the proposed method with U-attention-net and vanilla U-net under different pollution ratio in SQI dataset

	AUC(↑)	Acc(↑)	F1(↑)	TPR(↑)	FAR(↓)
			$\sigma = 0$		
U-attention-net	**1.00000 ± 0.00000**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**0.000% ± 0.000%**
Vanilla U-net	0.99999 ± 0.00002	99.957% ± 0.097%	99.957% ± 0.097%	99.957% ± 0.097%	0.086% ± 0.193%
			$\sigma = 0.1$		
U-attention-net	**0.99958 ± 0.00028**	**99.353% ± 0.305%**	**99.353% ± 0.305%**	**99.353% ± 0.305%**	0.603% ± 0.385%
Vanilla U-net	0.99923 ± 0.00051	99.224% ± 0.118%	99.224% ± 0.118%	99.224% ± 0.118%	**0.431% ± 0.305%**
			$\sigma = 0.2$		
U-attention-net	**0.99911 ± 0.00052**	**98.879% ± 0.515%**	**98.879% ± 0.515%**	**98.879% ± 0.515%**	**1.379% ± 1.029%**
Vanilla U-net	0.99855 ± 0.00037	98.491% ± 0.341%	98.491% ± 0.341%	98.491% ± 0.341%	1.724% ± 0.528%

components. The model retains the U-net structure but replaces the dilated convolutions with standard convolution and removes the self-attention block, denoted as vanilla U-net. The metrics under different contamination ratio are summarized in Tables III and IV for SQI and MGB dataset.

It can be seen that the U-attention-net outperforms the vanilla U-Net across most metrics and contamination levels. Specifically, in the SQI dataset, when polluted ratio σ is 0.2, U-attention-net maintains superior performance with an AUC of 0.99911 and FAR of 1.379%, compared to 0.99855 and 1.724% for the vanilla U-Net, respectively. On the MGB dataset, U-attention-net achieves an AUC of 0.98507 and FAR of 4.167%, while the vanilla U-Net attains a slightly lower AUC of 0.98299 and a higher FAR of 6.667%. These findings indicate that the

Table IV. Evaluation metrics of the proposed method with U-attention-net and vanilla U-net under different pollution ratio in MGB dataset

	AUC(↑)	Acc(↑)	F1(↑)	TPR(↑)	FAR(↓)
			$\sigma = 0$		
U-attention-net	**1.00000 ± 0.00000**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**100.000% ± 0.000%**	**0.000% ± 0.000%**
Vanilla U-net	1.00000 ± 0.00000	100.000% ± 0.000%	100.000% ± 0.000%	100.000% ± 0.000%	0.000% ± 0.000%
			$\sigma = 0.1$		
U-attention-net	**0.99999 ± 0.00003**	**99.917% ± 0.186%**	**99.917% ± 0.186%**	**99.917% ± 0.186%**	**0.167% ± 0.373%**
Vanilla U-net	0.99819 ± 0.00110	98.750% ± 0.417%	98.750% ± 0.417%	98.750% ± 0.417%	0.667% ± 0.697%
			$\sigma = 0.2$		
U-attention-net	**0.98507 ± 0.00920**	**95.500% ± 1.873%**	**95.499% ± 1.874%**	**95.500% ± 1.873%**	**4.167% ± 2.041%**
Vanilla U-net	0.98299 ± 0.00732	94.167% ± 1.271%	94.166% ± 1.271%	94.667% ± 1.271%	6.667% ± 1.728%

Table V. Evaluation metrics of the proposed method with and without filtered contrastive mechanism under different pollution ratio in SQI dataset

	AUC(↑)	Acc(↑)	F1(↑)	TPR(↑)	FAR(↓)
			$\sigma = 0.1$		
With FCM	**0.99958 ± 0.00028**	**99.353% ± 0.305%**	**99.353% ± 0.305%**	**99.353% ± 0.305%**	**0.603% ± 0.385%**
Without FCM	0.99941 ± 0.00025	99.267% ± 0.193%	99.267% ± 0.193%	99.267% ± 0.193%	0.690% ± 0.491%
			$\sigma = 0.2$		
With FCM	**0.99911 ± 0.00052**	**98.879% ± 0.515%**	**98.879% ± 0.515%**	**98.879% ± 0.515%**	**1.379% ± 1.029%**
Without FCM	0.99792 ± 0.00041	97.974% ± 0.327%	97.974% ± 0.327%	97.974% ± 0.327%	2.155% ± 0.747%

Table VI. Evaluation metrics of the proposed method with and without filtered contrastive mechanism under different pollution ratio in MGB dataset

	AUC(↑)	Acc(↑)	F1(↑)	TPR(↑)	FAR(↓)
			$\sigma = 0.1$		
With FCM	**0.99999 ± 0.00003**	**99.917% ± 0.186%**	**99.917% ± 0.186%**	**99.917% ± 0.186%**	**0.167% ± 0.373%**
Without FCM	0.94290 ± 0.01448	89.167% ± 2.412%	89.151% ± 2.423%	89.167% ± 2.412%	8.166% ± 3.084%
			$\sigma = 0.2$		
With FCM	**0.98507 ± 0.00920**	**95.500% ± 1.873%**	**95.499% ± 1.874%**	**95.500% ± 1.873%**	**4.167% ± 2.041%**
Without FCM	0.87881 ± 0.02594	80.833% ± 2.585%	80.789% ± 2.601%	80.833% ± 2.585%	17.333% ± 5.445%

incorporation of dilated convolution and self-attention mechanisms substantially boosts the model's performance to identify anomalies in high-dimensional sequences.

To assess the contribution of the FCM, an ablation study was conducted under contamination ratios 0.1 and 0.2. Tables V and VI report the performance of AD-DDPM with and without FCM, where the top-performing results are marked in bold. The inclusion of the FCM consistently improves performance across all metrics. It can be seen that the method with FCM performs better both in pollution rate 0.1 and 0.2. On the SQI dataset, for $\sigma = 0.1$, AD-DDPM with FCM achieves an FAR of 1.38%, compared to 2.16% without FCM. Similarly, on the MGB dataset at a pollution ratio of 0.2, the use of FCM leads to a substantial improvement in accuracy, reaching 95.50%, while the accuracy drops to 80.83% when FCM is removed.

The FCM's effectiveness stems from its dual components: filtering and contrastive learning. The filtering process identifies pseudo-anomalous samples by leveraging the higher anomaly score of anomalies during training, enabling the model to focus on normal patterns. The contrastive learning penalty further refines the latent representations by pulling normal sample features closer together while pushing anomalous features away, enhancing the model's discriminative power.

F. FLOPs AND TIME CONSUMPTION

Floating point operations (FLOPs) represent the total number of arithmetic operations performed by a model during a single forward pass. In general, models with higher FLOPs

Table VII. FLOPs, parameter amount, and train/infer time consumption of AE, VAE, and proposed AD-DDPM

Model name	FLOPs	Params	Train (s)	Infer (s)
AE-CNN	11.10G	15.00M	3.29	0.0053
VAE	11.16G	78.24M	3.31	0.0054
AD-DDPM	13.11G	22.33M	6.09	0.0046

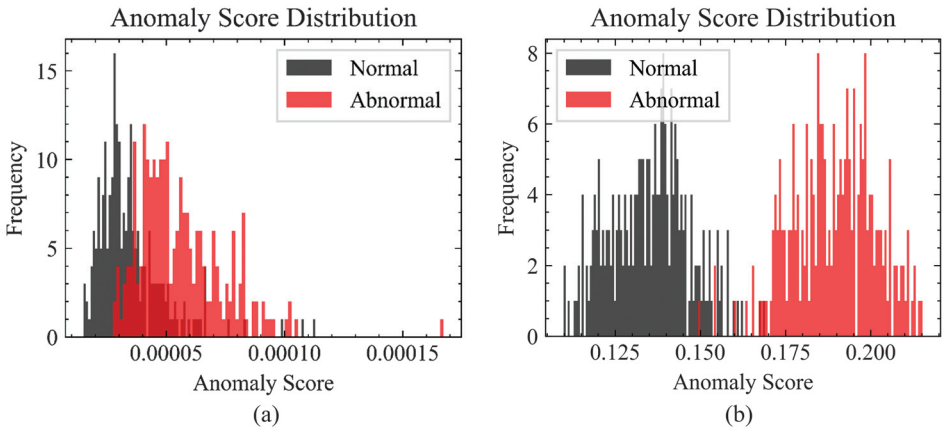

Fig. 5. The anomaly score distribution of (a) AE-CNN and (b) AD-DDPM.

require more computational resources and longer processing time. We report the FLOPs, parameter counts, training time per epoch, and inference time per sample for deep learning methods, as summarized in Table VII. Compared to AE and VAE, AD-DDPM incurs a slightly higher computational cost, with 13.11G FLOPs and 22.33M parameters. AD-DDPM requires a longer training time per epoch due to the additional overhead of updating pseudo-labels for FCM. It offers the fastest inference time as it computes anomaly scores directly from the noise difference without requiring inverse denoising, which is favorable for real-time deployment.

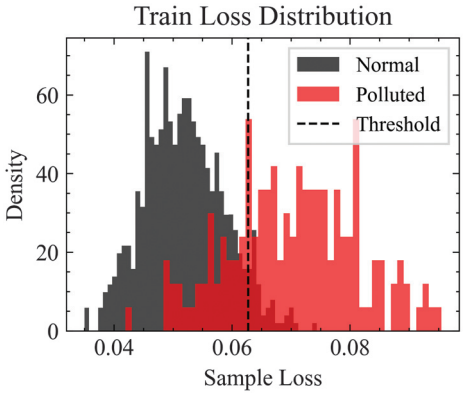

Fig. 6. The loss distribution at training epoch 30 with polluted data and the threshold setting.

G. VISUALIZATION

To provide intuitive insights into the AD-DDPM's performance, comprehensive visualizations were conducted in SQI dataset, encompassing anomaly score distributions, polluted sample loss distributions, signal reconstructions, feature embeddings, and generated signal comparisons.

Figure 5(a) and (b) compare the anomaly score distributions of AE-CNN and the proposed AD-DDPM models on the test dataset. The anomaly scores of AE model overlap significantly, indicating its weakness in differentiating anomalies and detecting anomalies. In contrast, AD-DDPM's anomaly scores are well separated. The scores of AE-CNN are mostly lower than 0.0001, suggesting it tends to overfit both normal and anomalous patterns, failing to capture fault-specific features, whereas AD-DDPM's probabilistic modeling of latent distributions ensures robust AD.

The loss distribution of polluted samples during training at epoch 30 is shown in Fig. 6. The distributions of normal and anomalous samples are distinctly separated, with anomalous samples exhibiting higher losses, shifted to the right of normal samples. This separation validates the efficacy of the FCM's threshold-based filtering, which accurately distinguishes pseudo-normal and pseudo-anomalous samples based on the quantile of the loss distribution. It accurately filters the probable pollution in the data and makes the model more robust.

During testing, an input sample is gradually noised over S steps to obtain X_i^S, which is then denoised through reverse steps to reconstruct $\widehat{X}_i^0$. Figure 7 visualizes the

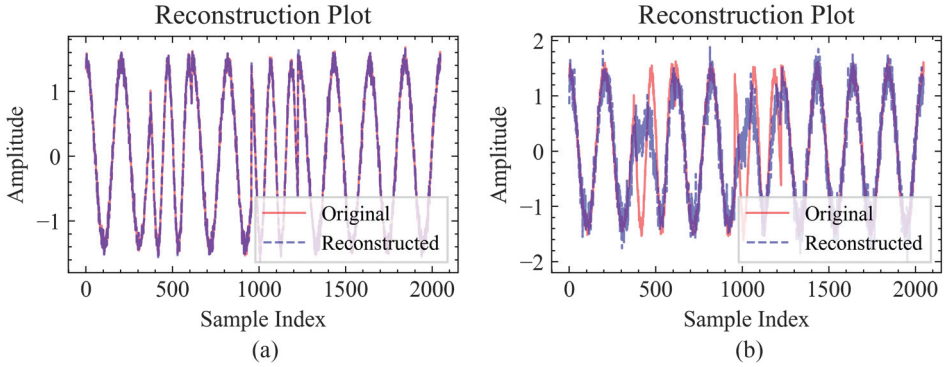

Fig. 7. Comparison of the original simulated abnormal signal with the reconstructed outputs from (a) AE and (b) AD-DDPM.

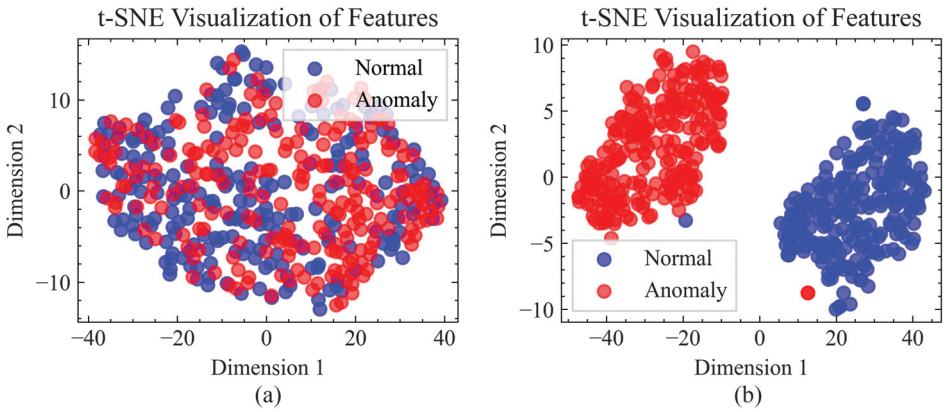

Fig. 8. The feature visualization after t-SNE dimension reduction of (a) AE and (b) AD-DDPM.

reconstruction of simulated signals by AD-DDPM compared to the AE model. The simulation signal is a sine signal with random Gaussian noise as the normal state. Random segments of sine signal's frequency are changed as the anomalies. The figure shows the abnormal signals. The AE model tends to fit every single detail including anomalous segments, thus failing to detect anomalies. In contrast, AD-DDPM reconstructs the inherent sine wave pattern, effectively repairing corrupted segments by leveraging the learned latent distribution and the generative reverse process of the diffusion model. This capability underscores AD-DDPM's ability to generalize normal patterns while identifying deviations as anomalies.

The feature embeddings after model bottleneck after t-SNE dimension reduction [43] are shown in Fig. 8. The AE model's features for normal and anomalous samples are heavily mixed, lacking a clear boundary, which reflects its limited discriminative power. AD-DDPM's feature map shows a clear separation between normal and anomalous samples, with normal features tightly clustered and anomalous features distinctly isolated. It demonstrates the method's capability to learn robust and discriminative representations, even under contaminated conditions.

To illustrate the diffusion model's generative capability, Fig. 9 compares generated vibration signals with real normal-state monitoring data, including their Fast Fourier Transform (FFT) spectrum. The generated signals exhibit frequency bands closely aligned with the input data, with minor differences in magnitude, indicating that AD-DDPM effectively captures the latent distribution of normal signals.

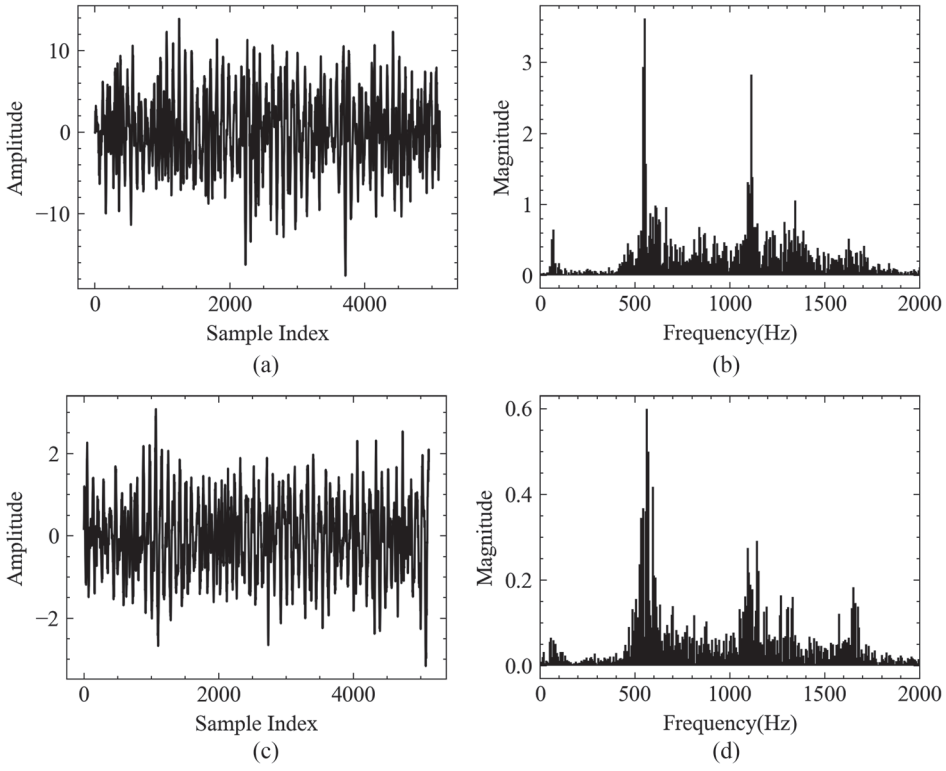

Fig. 9. The generated signals by AD-DDPM compared with collected vibration signals. (a) Time-domain vibration signal under normal conditions. (b) FFT spectrum of the normal signal. (c) Time-domain signal generated by DDPM. (d) FFT spectrum of the generated signal.

This generative ability not only validates the model's understanding of normal patterns but also highlights its potential for data augmentation in PHM applications, where labeled data is often scarce.

V. CONCLUSION

This paper presents a novel diffusion-based AD method, AD-DDPM, designed for robust monitoring of rotating machinery. DDPM probabilistically models the distribution of normal signals for accurate detection. The introduction of the U-attention-net, combining convolutional and self-attention mechanisms, enables the capture of both local and global patterns, boosting the model's capability to represent complex time-series data. Furthermore, the FCM effectively addresses the challenge of contaminated training data, ensuring robustness by distinguishing normal and anomalous samples through pseudo-labeling and contrastive learning. Experimental results on fault simulation datasets validate the superiority of AD-DDPM, achieving near-perfect detection performance across various contamination levels. Visualizations of anomaly scores, signal reconstructions, and feature embeddings further confirm the model's discriminative capability. Despite these promising results, several limitations should be acknowledged. First, the generative capability of the diffusion model has not yet been fully exploited for refining decision boundaries. Leveraging synthetic fault samples generated by the model could help clarify ambiguous regions in the feature space, especially under high contamination scenarios. Second, the experimental design considers limited operating conditions, which may challenge the model's generalization ability under multi-condition and variable-condition scenarios. Future research could address these issues by incorporating generative diffusion-based augmentation to refine decision boundaries and extending the method to multi-domain and cross-condition scenarios using domain adaptation or invariant representation learning to ensure robust generalization in real industrial systems.

ACKNOWLEDGMENTS

This project was supported by The National Natural Science Foundation of China under Grant (5247512) and National Key Lab of Aerospace Power System and Plasma Technology Foundation (APSPT202304002).

CONFLICT OF INTEREST STATEMENT

The authors declare no conflicts of interest.

REFERENCES

[1] W. Jiang, J. Wu, Y. Yang, X. Li, and H. Zhu, "Health evaluation techniques towards rotating machinery: a systematic literature review and implementation guideline," *Reliab. Eng. Syst. Saf.*, vol. 260, p. 110924, 2025.

[2] X. Liu, Z. Zhang, Z. Li, J. Wang, Y. Zhu, and H. Ma, "Advancements in bearing health monitoring and remaining useful life prediction: techniques, challenges, and future directions," *Meas. Sci. Technol.*, vol. 36, no. 3, p. 032003, 2025.

[3] F. Pittino, M. Puggl, T. Moldaschl, and C. Hirschl, "Automatic anomaly detection on in-production manufacturing machines using statistical learning methods," *Sensors*, vol. 20, no. 8, p. 8, 2020.

[4] E. Vanem and A. Brandsæter, "Unsupervised anomaly detection based on clustering methods and sensor data on a marine diesel engine," *J. Mar. Eng. Technol.*, vol. 20, no. 4, pp. 217–234, 2021.

[5] K. Vos, Z. Peng, C. Jenkins, M. R. Shahriar, P. Borghesani, and W. Wang, "Vibration-based anomaly detection using LSTM/SVM approaches," *Mech. Syst. Signal Process.*, vol. 169, p. 108752, 2022.

[6] M. Rao, M. J. Zuo, and Z. Tian, "A speed normalized autoencoder for rotating machinery fault detection under varying speed conditions," *Mech. Syst. Signal Process.*, vol. 189, p. 110109, 2023.

[7] Z. Zhao et al., "Model-driven deep unrolling: towards interpretable deep learning against noise attacks for intelligent fault diagnosis," *ISA Trans.*, vol. 129, pp. 644–662, 2022.

[8] A. Klausen, H. Van Khang, and K. G. Robbersmyr, "RMS based health indicators for remaining useful lifetime estimation of bearings," *Model. Identif. Control Nor. Res. Bull.*, vol. 43, no. 1, pp. 21–38, 2022.

[9] T. Yan, D. Wang, J.-Z. Kong, T. Xia, Z. Peng, and L. Xi, "Definition of signal-to-noise ratio of health indicators and its analytic optimization for machine performance degradation assessment," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–16, 2021.

[10] T. Yan, D. Wang, T. Xia, M. Zheng, Z. Peng, and L. Xi, "Entropy-maximization oriented interpretable health indicators for locating informative fault frequencies for machine health monitoring," *Mech. Syst. Signal Process.*, vol. 198, p. 110461, 2023.

[11] J. Qi, Z. Chen, Y. Uhlmann, and G. Schullerus, "Sensorless robust anomaly detection of roller chain systems based on motor driver data and deep weighted KNN," *IEEE Trans. Instrum. Meas.*, vol. 74, p. 3502613, 2025.

[12] C. Tutivén, Y. Vidal, A. Insuasty, L. Campoverde-Vilela, and W. Achicanoy, "Early fault diagnosis strategy for WT main bearings based on SCADA data and one-class SVM," *Energies*, vol. 15, no. 12, p. 4381, 2022.

[13] K. Shao, Y. He, Z. Xing, and B. Du, "Detecting wind turbine anomalies using nonlinear dynamic parameters-assisted machine learning with normal samples," *Reliab. Eng. Syst. Saf.*, vol. 233, p. 109092, 2023.

[14] Q. Xie, G. Tao, C. Xie, and Z. Wen, "Abnormal data detection based on adaptive sliding window and weighted multiscale local outlier factor for machinery health monitoring," *IEEE Trans. Ind. Electron.*, vol. 70, no. 11, pp. 11725–11734, 2023.

[15] Y. Lv, X. Guo, S. Shirmohammadi, L. Qian, Y. Gong, and X. Hu, "Intelligent cross-working condition fault detection and diagnosis using isolation forest and adversarial discriminant domain adaptation," *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–15, 2024.

[16] W. Li, Z. Shang, J. Zhang, M. Gao, and S. Qian, "A novel unsupervised anomaly detection method for rotating machinery based on memory augmented temporal convolutional autoencoder," *Eng. Appl. Artif. Intell.*, vol. 123, p. 106312, 2023.

[17] L. Yang and Z. Zhang, "Wind turbine gearbox failure detection based on SCADA data: a deep learning-based approach," *IEEE Trans. Instrum. Meas.*, vol. 70, pp. 1–11, 2021.

[18] H. J. Park, S. Kim, S.-Y. Han, S. Ham, K. J. Park, and J.-H. Choi, "Machine health assessment based on an anomaly indicator using a generative adversarial network," *Int. J. Precis. Eng. Manuf.*, vol. 22, no. 6, pp. 1113–1124, 2021.

[19] J. Wolleb, F. Bieder, R. Sandkühler, and P. C. Cattin, "Diffusion models for medical anomaly detection," In *Medical Image Computing and Computer Assisted Intervention – MICCAI 2022*, L. Wang, Q. Dou, P. T. Fletcher, S. Speidel, and S. Li, Eds. Cham: Springer Nature Switzerland, 2022, pp. 35–45.

[20] Z. Wang, X. Gu, J. Hu, and X. Gu, "Ensemble anomaly score for video anomaly detection using denoise diffusion model and motion filters," *Neurocomputing*, vol. 553, p. 126589, 2023.

[21] C. Wang et al., "Drift doesn't matter: dynamic decomposition with diffusion reconstruction for unstable multivariate time series anomaly detection," in *Proc. 37th Conf. Neural Inf. Process. Syst. (NeurIPS 2023)*, New Orleans, LA, USA, Dec. 2023, pp. 10758–10774.

[22] Y. Chen et al., "ImDiffusion: imputed diffusion models for multivariate time series anomaly detection," *Proc. VLDB Endow.*, vol. 17, no. 3, pp. 359–372, 2023.

[23] C. Yang, T. Wang, and X. Yan, "DDMT: denoising diffusion mask transformer models for multivariate time series anomaly detection," *arXiv preprint arXiv:2310.08800*, Oct. 2023.

[24] X. Xu, X. Yang, Z. Qiao, P. Liang, C. He, and P. Shi, "Multi-source domain adaptation using diffusion denoising for bearing fault diagnosis under variable working conditions," *Knowl.-Based Syst.*, vol. 302, p. 112396, 2024.

[25] J. Yoon, K. Sohn, C.-L. Li, S. O. Arik, C.-Y. Lee, and T. Pfister, "Self-trained one-class classification for unsupervised anomaly detection," *arXiv preprint arXiv:2106.06115*, Jun. 2021.

[26] M. Ulmer, J. Zgraggen, and L. G. Huber, "A generic machine learning framework for fully-unsupervised anomaly detection with contaminated data," *Int. J. Progn. Health Manag.*, vol. 15, no. 1, pp. 1–12, 2024.

[27] B. Du, X. Sun, J. Ye, K. Cheng, J. Wang, and L. Sun, "GAN-based anomaly detection for multivariate time series using polluted training set," *IEEE Trans. Knowl. Data Eng.*, vol. 35, no. 12, pp. 12208–12219, 2023.

[28] Z. Shang, Z. Zhao, R. Yan, and X. Chen, "Core loss: Mining core samples efficiently for robust machine anomaly detection against data pollution," *Mech. Syst. Signal Process.*, vol. 189, p. 110046, 2023.

[29] C. Qiu, A. Li, M. Kloft, M. Rudolph, and S. Mandt, "Latent outlier exposure for anomaly detection with contaminated data," in *Proc. 39th Int. Conf. Mach. Learn. (ICML)*, PMLR, Jun. 2022, pp. 18153–18167.

[30] X. Mou et al., "RoCA: robust contrastive one-class time series anomaly detection with contaminated data," *arXiv preprint arXiv:2503.18385*, 2025.

[31] Q. Su, B. Tian, H. Wan, and J. Yin, "Anomaly detection under contaminated data with contamination-immune bidirectional GANs," *IEEE Trans. Knowl. Data Eng.*, vol. 36, no. 11, pp. 5605–5620, 2024.

[32] C. Zhou and R. C. Paffenroth, "Anomaly detection with robust deep autoencoders," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discov. Data Min. (KDD '17)*, New York, NY, USA: ACM, Aug. 2017, pp. 665–674.

[33] S. Wang et al., "E3 outlier: a self-supervised framework for unsupervised deep outlier detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 45, no. 3, pp. 2952–2969, 2023.

[34] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *Adv. Neural Inf. Process. Syst.*, vol. 33, pp. 6840–6851, 2020.

[35] O. Ronneberger, P. Fischer, and T. Brox, "U-net: convolutional networks for biomedical image segmentation," In *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015: 18th International Conference, Munich, Germany, October 5–9, 2015, Proceedings, Part III*, vol. 9351. Cham: Springer, 2015, pp. 234–241.

[36] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, "FiLM: visual reasoning with a general conditioning Layer," *Proc. AAAI Conf. Artif. Intell.*, vol. 32, no. 1, 2018.

[37] A. Vaswani et al., "Attention is all you need," *Adv. Neural Inf. Process. Syst.*, vol. 30, 2017.

[38] Z. Wang, Y. Fu, C. Song, P. Zeng, and L. Qiao, "Power system anomaly detection based on OCSVM optimized by improved particle swarm optimization," *IEEE Access*, vol. 7, pp. 181580–181588, 2019.

[39] S. Zhong, S. Fu, L. Lin, X. Fu, Z. Cui, and R. Wang, "A novel unsupervised anomaly detection for gas turbine using isolation forest," in *Proc. IEEE Int. Conf. Prognostics Health Manag. (ICPHM)*, Jun. 2019, pp. 1–6.

[40] Z. Cheng, C. Zou, and J. Dong, "Outlier detection using isolation forest and local outlier factor," in *Proc. Conf. Research in Adaptive and Convergent Systems (RACS '19)*, New York, NY, USA: ACM, Sep. 2019, pp. 161–168.

[41] J. Wu, C. Hu, C. Sun, Z. Zhao, R. Yan, and X. Chen, "Helicopter transmission system anomaly detection in variable flight regimes with decoupling variational autoencoder," *Aerosp. Sci. Technol.*, vol. 144, p. 108764, 2024.

[42] W. Guan, J. Cao, Y. Gu, and S. Qian, "GRASPED: a GRU-AE network based multi-perspective business process anomaly detection model," *IEEE Trans. Serv. Comput.*, vol. 16, no. 5, pp. 3412–3424, 2023.

[43] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE.," *J. Mach. Learn. Res.*, vol. 9, no. 11, pp. 2579–2605, 2008.